



BAB II

TINJAUAN PUSTAKA

2.1. Teori Umum

2.1.1. Pengertian Komputer

Asropudin (2013:19), “Komputer adalah alat bantu pemrosesan data secara elektronik dan cara pemrosesan datanya berdasarkan urutan instruksi atau program yang tersimpan dalam memori masing-masing”.

Sutarman (2012:3), “Komputer adalah alat elektronik yang dapat melakukan perhitungan numerik, dapat membaca *input* data dan mengolahnya sesuai dengan program yang ditetapkan untuk menghasilkan sebuah *output*, alat yang digunakan untuk menyimpan data, serta bekerja secara otomatis sesuai dengan perintah yang diterapkan dalam program”.

Jadi komputer merupakan sebuah alat bantu dalam mengolah data elektronik yang dapat melakukan pekerjaan melalui instruksi-instruksi tertentu yang akan menghasilkan informasi dan mempermudah dalam membantu pengolahan data.

2.1.2. Pengertian Program

Sutarman (2012:3), “Program adalah barisan perintah/instruksi yang disusun sehingga dapat dipahami oleh komputer dan kemudian dijalankan sebagai barisan perhitungan numerik, di mana barisan perintah tersebut berhingga, berakhir, dan menghasilkan *output*”.

2.1.3. Pengertian Sistem

Sutarman (2012:5), “Sistem adalah kumpulan elemen yang saling berhubungan dan berinteraksi dalam satu kesatuan untuk menjalankan suatu proses pencapaian suatu tujuan utama”.

Hartono (2013:10), sistem yakni suatu benda atau entitas yaitu himpunan dari berbagai bagian atau komponen, dan sekaligus juga suatu proses atau metode atau cara untuk mencapai tujuan yaitu saling berhubungan secara terorganisasi berdasar fungsi-fungsinya.



2.1.4. Pengertian Informasi

Hartono (2013:15), Informasi adalah sehimpunan data yang telah diolah menjadi sesuatu yang memiliki arti dan kegunaan lebih luas.

Ladjamudin (2013:8), Informasi adalah data yang telah diolah menjadi bentuk yang lebih berarti dan berguna bagi penerimanya untuk mengambil keputusan masa kini maupun yang akan datang.

2.1.5. Pengertian Sistem Informasi

Sutarman (2012:13), Sistem informasi adalah sistem yang mengumpulkan, memproses, menyimpan, menganalisis, menyebarkan *input* dan menghasilkan *output* menjadi informasi untuk tujuan tertentu kepada pengguna atau sistem yang lainnya.

Hartono (2013:20), “Sistem informasi adalah rangkaian sistem yang terorganisasi dari sejumlah bagian/komponen yang secara bersama-sama berfungsi atau bergerak menghasilkan informasi”.

2.1.6. Pengertian Pangkat

Fokusmedia (2014:175), “Pangkat adalah kedudukan yang menunjukkan tingkatan seseorang Pegawai Negeri Sipil berdasarkan jabatannya dalam rangkaian susunan kepegawaian dan digunakan sebagai dasar penggajian”.

2.1.7. Pengertian Kenaikan Pangkat

Fokusmedia (2014:175), “Kenaikan pangkat adalah penghargaan yang diberikan atas prestasi kerja dan pengabdian Pegawai Negeri Sipil terhadap Negara, serta sebagai dorongan kepada Pegawai Negeri Sipil untuk lebih meningkatkan prestasi kerja dan pengabdianya”.

2.1.8. Pengertian Pegawai Negeri Sipil

Fokusmedia (2014:3), “Pegawai Negeri Sipil (PNS) adalah warga Negara Indonesia yang memenuhi syarat tertentu, diangkat sebagai pegawai ASN secara tetap oleh pejabat pembina kepegawaian untuk menduduki jabatan pemerintahan”.

2.1.9. Pengertian Sistem Informasi Kenaikan Pangkat Pegawai Negeri Sipil pada Kejaksaan Tinggi Sumatera Selatan

Sistem Informasi Kenaikan Pangkat Pegawai Negeri Sipil pada Kejaksaan Tinggi Sumatera Selatan adalah sebuah sistem informasi yang mengolah data kenaikan pangkat untuk menghasilkan laporan hasil kenaikan pangkat pegawai negeri sipil pada Kejaksaan Tinggi Sumatera Selatan.

2.2. Teori Khusus



2.2.1. Data Flow Diagram (DFD)

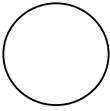
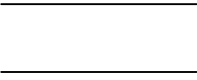
Ladjamudin (2013:64), “*Data flow diagram* atau diagram aliran data (DFD) merupakan model dari sistem untuk menggambarkan pembagian sistem ke modul yang lebih kecil”.

Shalahuddin et.al (2013:70), “*Data flow diagram* (DFD) adalah representasi grafik yang menggambarkan aliran informasi dan transformasi yang diaplikasikan sebagai data yang mengalir dari masukan (*input*) dan keluaran (*output*)”.

Ada beberapa simbol DFD yang dipakai untuk menggambarkan data beserta proses transformasi data, antara lain:

Tabel 2.1
Simbol- Simbol *Data Flow Diagram*

Simbol	Keterangan
	<i>Entitas</i> atau masukan (<i>input</i>) atau keluaran (<i>output</i>) atau orang. Yaitu merupakan kesatuan di lingkungan luar sistem yang dapat berupa orang, organisasi atau sistem lainnya yang berada di lingkungan luarnya yang akan memberikan <i>input</i> atau menerima <i>output</i> dari sistem.
	Aliran data; merupakan data yang dikirim antarproses, dari penyimpanan ke proses atau dari proses ke masukan (<i>input</i>) atau keluaran (<i>output</i>). Nama yang diberikan biasanya berupa kata benda, dapat diawali dengan kata data misalnya “data_siswa” atau tanpa kata data misalnya “siswa”.

Simbol	Keterangan
	Proses atau fungsi atau prosedur. Simbol ini merupakan fungsi atau prosedur di dalam kode program. Nama pada proses berupa kata kerja.
	<i>Field</i> atau basis data atau penyimpanan (<i>storage</i>). Simbol ini harus dibuat menjadi tabel penyimpanan yang dibutuhkan dan sesuai dengan perancangan. Nama pada penyimpanan berupa kata benda.

(Sumber: Shalahuddin *et al*, 2013:71)

2.2.2. Kamus Data (*Data Dictionary*)

Shalahuddin *et al* (2013:73), “Kamus data (*data dictionary*) adalah kumpulan daftar elemen data yang mengalir pada sistem sehingga masukan dan keluaran dapat dipahami secara umum (memiliki standar cara penulisan)”.

Adapun simbol-simbol dalam kamus data adalah sebagai berikut:

Tabel 2.2

Simbol-simbol Kamus Data (*Data Dictionary*)

Simbol	Keterangan
=	Disusun atau terdiri dari
+	Dan
()	Data opsional
[]	Baik ... atau ...
...	Batas komentar
{ } ⁿ	N kali diulangi/bernilai banyak

(Sumber: Shalahuddin *et al*, 2013:74)

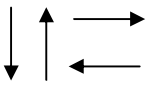
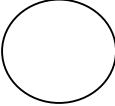
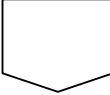


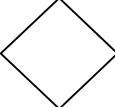

2.2.3. Pengertian *Flowchart*



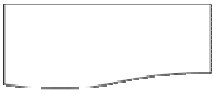

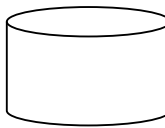
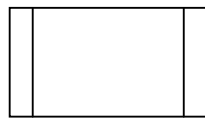
Ladjudin (2013:263), “*Flowchart* adalah bagan-bagan yang mempunyai arus yang menggambarkan langkah-langkah penyelesaian suatu masalah”.

eWolf (2012:16), “*Flowchart* adalah simbol-simbol pekerjaan yang menunjukkan bagan aliran proses yang saling terhubung. Jadi setiap simbol *flowchart* melambangkan pekerjaan dan instruksinya. *Flowchart* digunakan untuk mempermudah penyusunan program. Dengan menggunakan *flowchart*, logika pemrograman lebih dapat dipahami dan dianalisis, sehingga dapat menentukan kode-kode pemrograman yang sesuai dengan pekerjaannya”.

Berikut adalah beberapa simbol standar yang sering digunakan dalam pemrograman komputer, yaitu:

Tabel 2.3
Simbol-simbol *Flowchart*

Simbol	Nama Simbol	Keterangan
	Garis Alir (<i>Flow Line</i>)	Simbol untuk menghubungkan antar proses atau antar simbol
	<i>Connector</i>	Simbol konektor untuk menyambung proses pada lembar kerja yang sama
	<i>Offline Connector</i>	Simbol konektor untuk menyambung proses pada lembar kerja yang berbeda
	<i>Process</i>	Simbol pemrosesan yang terjadi pada sebuah alur kerja
	<i>Manual</i>	Simbol yang mendefinisikan proses yang dilakukan secara manual
	<i>Decision</i>	Simbol untuk memutuskan proses lanjutan dari kondisi tertentu
	<i>Terminal</i>	Simbol <i>start</i> atau <i>end</i> yang mendefinisikan awal atau akhir dari sebuah <i>flowchart</i>

Simbol	Nama Simbol	Keterangan
	<i>Input/Output</i>	Simbol <i>input/output</i> yang mendefinisikan masukan dan keluaran proses
	<i>Magnetic Tape</i>	Simbol masukan atau keluaran dari atau ke sebuah pita magnetik
	<i>Document</i>	Simbol masukan atau keluaran dari atau ke sebuah dokumen
	<i>Display</i>	Simbol yang menyatakan piranti keluaran, seperti layar monitor, printer, dll
	<i>Data Storage</i>	Simbol database atau basis data
	<i>Sub Program</i>	Simbol yang menyatakan bagian dari program (sub program)


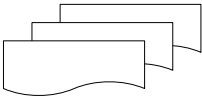
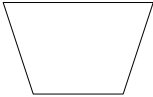

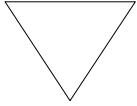
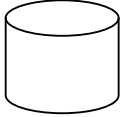

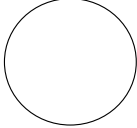
(Sumber: eWolf Community, 2012:17)


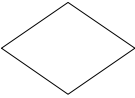
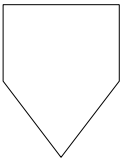
2.2.4. Blockchart

Kristanto (2008:75), “*Blockchart* berfungsi untuk memodelkan masukan, keluaran, proses maupun transaksi dengan menggunakan simbol-simbol tertentu. Pembuatan *blockchart* harus memudahkan bagi pemakai dalam memahami alur dari sistem atau transaksi”.

Adapun simbol-simbol digunakan dalam *blockchart* adalah sebagai berikut:

Tabel 2.4
Simbol-simbol *blockchart*

No.	Nama Simbol	Simbol	Arti
1.	<i>Document</i>		Simbol yang digunakan dalam menandakan suatu dokumen, bisa dalam bentuk surat, formulir, berkas atau cetakan.
2.	<i>Multi Document</i>		Suatu simbol yang digunakan dalam menandakan suatu multi dokumen, bisa dalam bentuk surat, formulir, berkas atau cetakan.
3.	<i>Manual Operation</i>		Suatu simbol yang digunakan dalam suatu proses yang dilakukan secara manual.
4.	<i>Process</i>		Suatu proses yang dilakukan dengan menggunakan komputer.
5.	<i>Merge</i>		Simbol yang digunakan dalam dokumen yang menandakan dokumen diarsipkan (arsip manual).
6.	<i>Magnetik Disk</i>		Suatu simbol yang digunakan dalam data penyimpanan (<i>data storage</i>).
7.	<i>Tampilan</i>		Layar peraga (monitor)
8.	<i>Connector</i>		Suatu simbol yang digunakan dalam terminasi yang mewakili simbol-simbol tertentu untuk digunakan pada aliran lain pada halaman yang sama.

No.	Nama Simbol	Simbol	Arti
9.	<i>Alternate Process</i>		Simbol yang digunakan dalam terminasi yang menandakan awal; atau akhir dari suatu aliran.
10.	<i>Decision</i>		Suatu simbol yang digunakan dalam pengambilan keputusan akhir.
11.	<i>Off-Page Connector</i>		Suatu simbol yang digunakan dalam terminasi yang mewakili simbol-simbol tertentu untuk digunakan pada aliran lain pada halaman yang lain.

(Sumber: Kristanto, 2008:75)

2.2.5. Entity Relational Diagram (ERD)

Ladjamudin (2013:142), “*Entity Relationship Diagram/ERD* adalah suatu model jaringan yang menggunakan susunan data yang disimpan dalam sistem secara bastrak. Jadi jelaslah bahwa ERD ini berbeda dengan DFD yang merupakan suatu model jaringan fungsi yang akan dilaksanakan oleh sistem, sedangkan ERD merupakan model jaringan data yang menekankan pada struktur-struktur dan *relationship* data”.


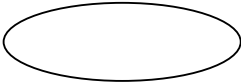
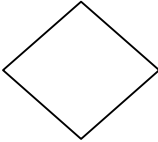

Shalahuddin et.al (2013:50), “*Entity Relationship Diagram (ERD)* adalah pemodelan awal basis data yang dikembangkan berdasarkan teori himpunan dalam bidang matematika untuk pemodelan basis data relasional”.

Yakub (2012:60), “*Entity Relationship Diagram (ERD)* merupakan suatu model jaringan yang menggunakan susunan data yang disimpan pada sistem secara abstrak. *ERD* menggambarkan hubungan antara satu entitas yang memiliki sejumlah atribut dengan entitas yang lain dalam suatu sistem yang terintegrasi”.

Adapun simbol-simbol *Entity Relationship Diagram* adalah sebagai berikut:

Tabel 2.5

Simbol- Simbol *Entity Relationship Diagram* (ERD)

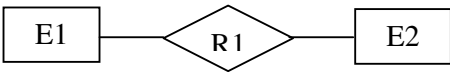
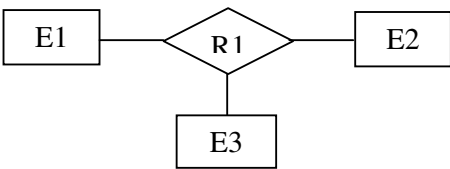
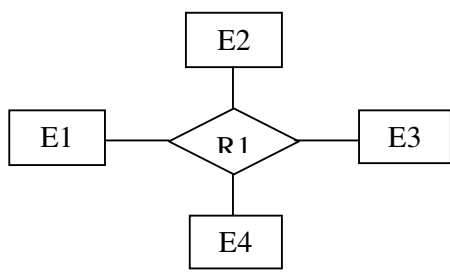
Simbol	Deskripsi
Entitas / Entity 	Kumpulan dari objek yang dapat diidentifikasi secara unik dan menunjukkan objek-objek dasar yang terkait di dalam sistem.
Atribut 	Karakteristik dari entitas atau relasi yang merupakan penjelasan detail tentang entitas yang perlu disimpan sebagai basis data.
Relasi/Relation 	Hubungan yang terjadi antara satu atau lebih entitas. Jenis hubungan antara lain: satu ke satu, satu ke banyak, dan banyak ke banyak.
Link 	Hubungan antara entitas dengan atributnya dan himpunan entitas dengan himpunan relasinya.

(Sumber: Yakub, 2012:60)

ERD biasanya memiliki hubungan *binary* (satu relasi menghubungkan data dua buah entitas). Beberapa metode perancangan ERD menoleransi hubungan relasi *tenary* (satu relasi menghubungkan tiga buah relasi) atau *N-ary* (satu relasi menghubungkan banyak entitas), tapi banyak metode perancangan ERD yang tidak mengizinkan hubungan ternary atau *N-ary*. Berikut adalah contoh bentuk hubungan relasi dalam ERD.

Tabel 2.6

Hubungan Relasi dalam *Entity Relationship Diagram* (ERD)

Nama	Gambar
<i>Binary</i>	
<i>Ternary</i>	
<i>N-ary</i>	

(Sumber: Shalahuddin et.al, 2013:62)

2.3. Teori Program

2.3.1. PHP (*Hypertext Preprocessor*)

2.3.1.1. Pengertian *PHP*

Madcoms (2011:216), “PHP adalah salah satu bahasa pemrograman yang berjalan dalam sebuah *webserver* dan berfungsi sebagai pengolah data pada sebuah *server*”.

2.3.1.2. Penulisan *Script PHP*

Wahana Komputer (2010:1), “Seperti *web* yang lain, baik sisi *client* maupun *server*, *PHP* juga mempunyai *tag* awal (pembuka) dan *tag* akhir (penutup) untuk mengenali bahwasanya *script PHP* telah ditulis pada suatu halaman *web*. Terdapat 4 cara untuk memulai menuliskan *script PHP*, seperti *source code* berikut:



Source code: tag awal dan akhir *PHP* model pertama

```
<?php
    Echo "Tag awal dan akhir menuliskan script PHP model ke-1
    &lt;?php...?&gt;";
?>
```

Source code: Tag awal dan akhir *PHP* model kedua

```
<?
    Echo "Tag awal dan akhir menuliskan script PHP model ke-2
    &lt;?...?&gt;";
?>
```

Source code: Tag awal dan akhir model ketiga

```
<script language="php">
    Echo "perintah model ketiga &lt;script language="php"&gt;
    ...&lt;/script&gt;";
</script>
```

Source code: Tag awal dan akhir model keempat

```
<%
    Echo "Tag awal dan akhir menuliskan script PHP model ke-4
    &lt;%...%&gt;";
%>
```

2.3.2. Adobe Dreamweaver CS5

2.3.2.1. Pengertian Dreamweaver

Madcoms (2011:2), "*dreamweaver* adalah sebuah *HTML editor professional* untuk mendesain *web* secara *visual* dan mengelola situs atau halaman *web*".

2.3.2.2. Tampilan Awal



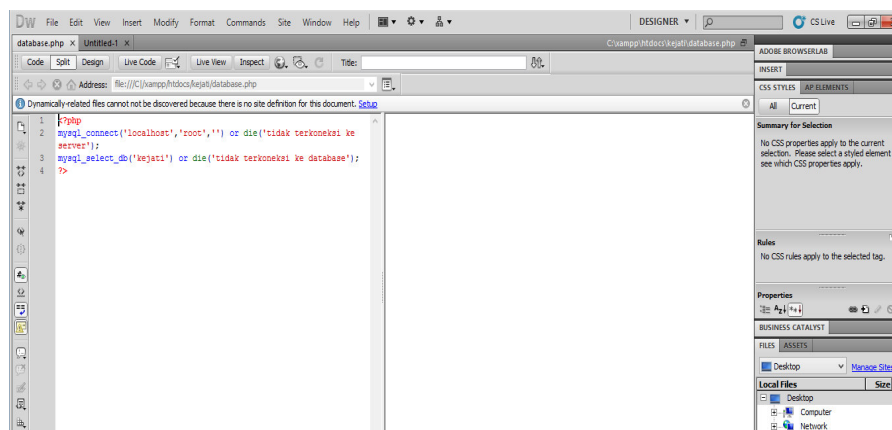
Gambar 2.1

Tampilan awal *Dreamweaver*

Dalam tampilan awal *Dreamweaver* terdapat pilihan *Open a Recent Item* (file yang pernah terbuka), *Create New* (buat file baru), dan *Getting Started* (tuntunan menggunakan *Dreamweaver*). Halaman *Welcome Screen* akan selalu ditampilkan saat program *Dreamweaver* dijalankan.

2.3.2.3. Ruang Kerja *Dreamweaver*

Tampilan ruang kerja *Dreamweaver* CS5 seperti yang terlihat pada gambar dibawah ini :



Gambar 2.2

Ruang kerja *Dreamweaver*



1. Application Bar

Berada di bagian paling atas jendela aplikasi *Dreamweaver* CS5. Baris ini berisi *workspace* (*workspace switcher*), menu, dan aplikasi lainnya.

2. Toolbar Document

Berisi tombol-tombol yang digunakan untuk mengubah tampilan jendela dokumen, sebagai contoh tampilan *Design* atau tampilan *Code*.

3. Jendela Document

Adalah lembar kerja tempat membuat dan mengedit *design* halaman *web*.

4. Workspace Switcher

Digunakan untuk mengubah tampilan ruang kerja (*workspace*) *Dreamweaver* CS5.

5. Panel Groups

Adalah kumpulan panel yang saling berkaitan, panel-panel ini dikelompokkan pada judul-judul tertentu berdasarkan fungsinya. Panel ini digunakan untuk memonitor dan memodifikasi pekerjaan.

6. Tag Selector

Diletakkan di bagian bawah jendela dokumen, satu baris dengan *status bar*. Bagian ini menampilkan hirarki pekerjaan yang sedang terpilih pada jendela dokumen. Dapat juga digunakan untuk memilih objek pada jendela desain berdasarkan jenis atau kategori objek tersebut.

7. Panel Properti

Digunakan untuk melihat dan mengubah berbagai *property* objek atau teks pada jendela *design*.

8. Toolbar Coding

Berisi tombol-tombol yang digunakan untuk melakukan operasi *code-code* standar. *Toolbar* ini hanya tampil pada jendela *Code*.

9. Panel Insert

Berisi tombol-tombol untuk menyisipkan berbagai jenis objek, seperti *image*, *table*, atau objek media ke dalam jendela dokumen. Masing-masing objek adalah suatu potongan *code HTML*.



10. Panel Files

Digunakan untuk mengatur *file-file* dan *folder-folder* yang membentuk situs *Web*. Selain itu, panel *files* dapat digunakan juga untuk melakukan *upload file* ke *server*, baik *server intranet* maupun *internet*.

2.3.3. MySQL

2.3.3.1. Pengertian MySQL

Anhar (2010:21), *MySQL (My Structure Query Language)* adalah sebuah perangkat lunak sistem manajemen basis data *SQL (Database Management System)* atau *DBMS* dari sekian banyak *DBMS*, seperti *Oracle*, *MS SQL*, *Postagre SQL*, dan lain-lain.

Wahana (2010:5), *MySQL* adalah program *database* yang mampu mengirim dan menerima data dengan sangat cepat dan *multi-user*.

2.3.3.2. Fungsi-Fungsi MySQL

Untuk dapat menggunakan semua perintah *SQL* pada *PHP*, kita memerlukan fungsi *MySQL*. Fungsi ini berguna untuk mengantarkan perintah *SQL* pada *PHP* menuju ke *server* sehingga perintah tersebut dapat dieksekusi oleh semua *server MySQL*.

Fungsi-fungsi *MySQL* yang berawalan *mysql_* yang digunakan untuk mengakses *database server MySQL* adalah sebagai berikut:

Tabel 2.7
Fungsi-fungsi *MySQL* dalam *PHP*

No.	Fungsi MySQL	Penggunaan
1	<i>Mysql_connect()</i>	Membuat hubungan ke <i>database MySQL</i> yang terdapat pada suatu host
2	<i>Mysql_close()</i>	Menutup hubungan ke <i>database MySQL</i>
3	<i>Mysql_select_db()</i>	Memilih <i>database</i>



No.	Fungsi MySQL	Penggunaan
4	<i>Mysql_query()</i>	Mengeksekusi permintaan terhadap sebuah tabel atau sejumlah tabel
5	<i>Mysql_db_query()</i>	Menjalankan suatu permintaan terhadap suatu <i>database</i>
6	<i>Mysql_num_rows()</i>	Memperoleh jumlah baris dari suatu hasil permintaan yang menggunakan <i>SELECT</i>
7	<i>Mysql_affected_rows()</i>	Memperoleh jumlah baris yang dikenai operasi <i>INSERT, DELETE, UPDATE</i>
8	<i>Mysql_num_fields()</i>	Memperoleh jumlah kolom pada suatu hasil permintaan
9	<i>Mysql_fetch_row()</i>	Menghasilkan <i>array</i> /baris yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasil
10	<i>Mysql_fetch_array()</i>	Menghasilkan <i>array</i> /baris yang berisi seluruh kolom dari sebuah baris pada suatu himpunan hasil yang akan disimpan dua kali pada <i>array</i> hasil
11	<i>Mysql_fetch_field()</i>	Menghasilkan informasi suatu kolom
12	<i>Mysql_data_seek()</i>	Memindahkan pointer pada suatu himpunan hasil supaya menunjuk ke baris tertentu
13	<i>Mysql_field_seek()</i>	Memindahkan pointer pada suatu himpunan hasil supaya menunjuk ke kolom tertentu
14	<i>Mysql_create_db()</i>	Membuat <i>database MySQL</i>
15	<i>Mysql_drop_db()</i>	Menghapus <i>database MySQL</i>
16	<i>Mysql_list_dbs()</i>	Menghasilkan daftar <i>database MySQL</i>
17	<i>Mysql_list_tables()</i>	Memperoleh daftar nama tabel dalam suatu <i>database</i>
18	<i>Mysql_list_fields()</i>	Memperoleh daftar nama kolom dalam suatu <i>database</i>



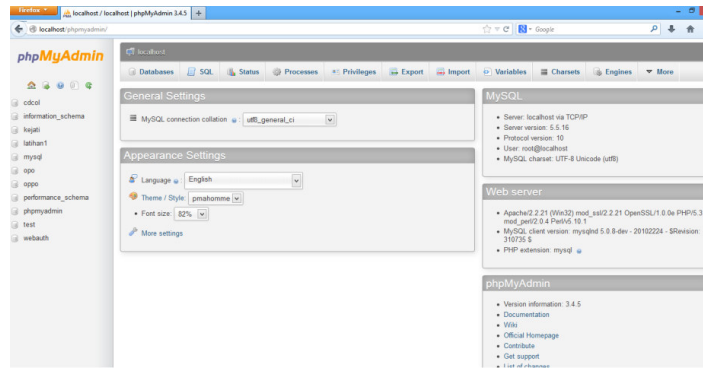
No.	Fungsi MySQL	Penggunaan
19	<i>Mysql_fetch_assoc()</i>	Mendapatkan array baris dari suatu <i>recordset</i>
20	<i>Mysql_fetch_lengths()</i>	Mendapatkan panjang baris pada setiap isi <i>field</i>
21	<i>Mysql_fetch_object()</i>	Menghasilkan baris dari <i>recordset</i> sebagai sebuah objek
22	<i>Mysql_field_len()</i>	Mendapatkan informasi panjang maksimum <i>field</i> dalam sebuah <i>recorset</i>
23	<i>Mysql_field_name()</i>	Mendapatkan informasi nama <i>field</i> dalam <i>recordset</i>
24	<i>Mysql_get_client_info()</i>	Mendapatkan informasi tentang <i>MySQL client</i>
25	<i>Mysql_get_host_info()</i>	Mendapatkan informasi tentang <i>MySQL client</i>
26	<i>Mysql_get_server_info()</i>	Mendapatkan informasi tentang <i>MySQL server</i>
27	<i>Mysql_info()</i>	Mendapatkan informasi tentang <i>query</i> terakhir
28	<i>Mysql_ping()</i>	Memeriksa koneksi <i>server</i> dan akan mencoba untuk melakukan koneksi ulang jika koneksi terputus
29	<i>Mysql_result()</i>	Mendapatkan nilai dari <i>field</i> dalam sebuah <i>recordset</i>
30	<i>Mysql_stat()</i>	Menghasilkan informasi status <i>server</i> terkini

(Sumber: eWolf Community (2012:157-168))

2.3.3.3. Langkah-Langkah Membuat Database dan Tabel

Madcoms (2011:290), untuk membuat *website* yang dinamis, pertama kali yang perlu disiapkan adalah sebuah *database* sebagai tempat untuk menyimpan data. Fungsi yang digunakan untuk membuat *database* ini adalah *mysql_create_db()*.

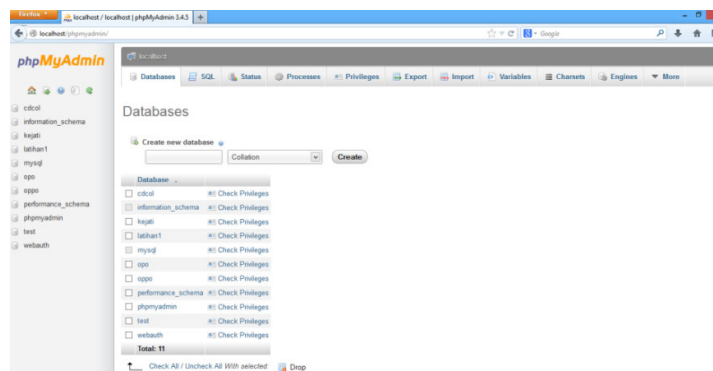
1. Jalankan jendela browser, dan masukkan alamat untuk membuka *phpMyAdmin*. Sebagai contoh penulis menggunakan alamat *localhost/sql*.



Gambar 2.3

Tampilan <http://localhost/phpmyadmin/>

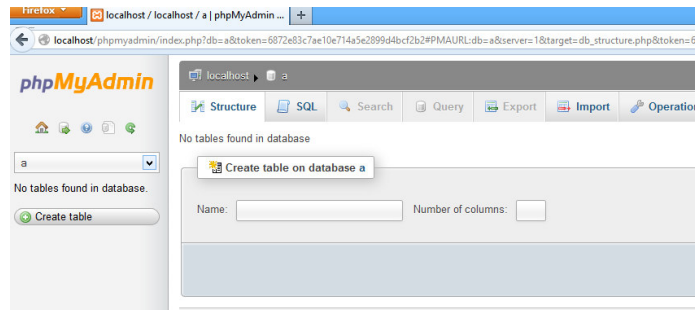
2. Ketik nama *file database* pada kotak *Create new database* misalnya *kejati*, kemudian tekan *create*.



Gambar 2.4

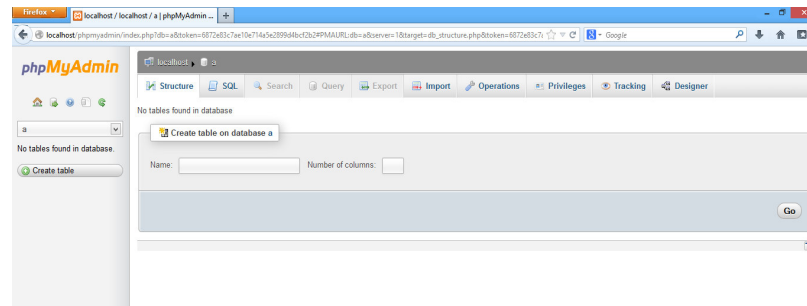
Tampilan membuat *Database*

3. Berikutnya akan tampil halaman untuk membuat tabel baru. Klik pada bagian **Database** untuk menguji apakah *file database* yang dibuat telah selesai.

**Gambar 2.5**

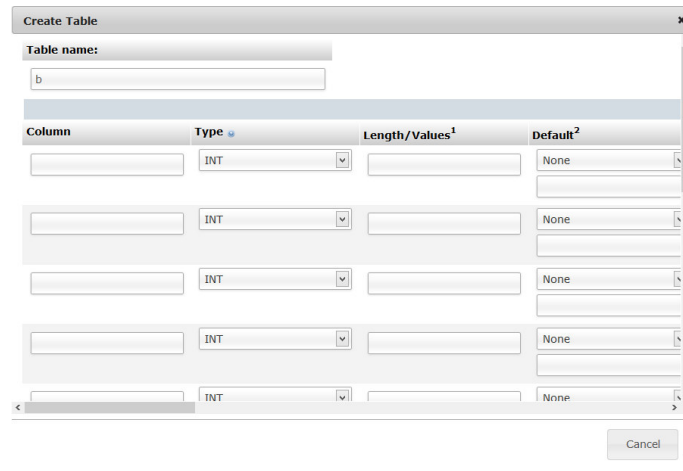
Tampilan membuat tabel baru

4. Setelah membuat *database*, ketik nama tabel pada bagian **Create table on**. Tentukan jumlah *field*-nya, kemudian klik tombol **Go**.

**Gambar 2.6**

Tampilan menentukan *field* pada *database*

5. Pada langkah berikutnya isi **field**, **tipe data**, dan informasi lainnya.

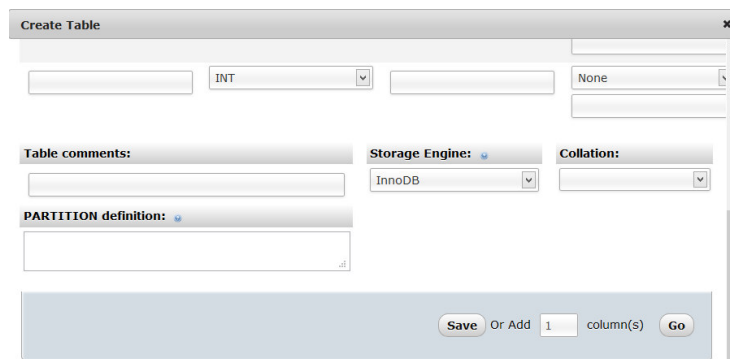


The screenshot shows a 'Create Table' dialog box. The 'Table name' field contains 'b'. Below it is a table structure with 5 columns. Each column has a text input field, a 'Type' dropdown set to 'INT', a 'Length/Values' input field, and a 'Default' dropdown set to 'None'. A 'Cancel' button is at the bottom right.

Column	Type	Length/Values ¹	Default ²
	INT		None
	INT		None
	INT		None
	INT		None
	INT		None

Gambar 2.7
Tampilan *Field* Tabel

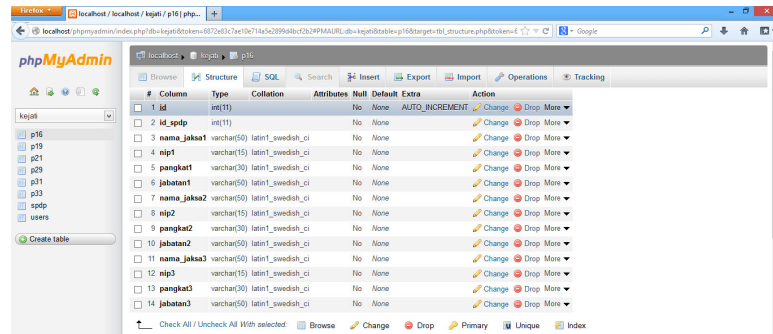
6. Klik tombol **save** untuk menyimpan tabel.



The screenshot shows the 'Create Table' dialog box with additional options. The 'Table comments' field is empty. The 'Storage Engine' dropdown is set to 'InnoDB'. The 'Collation' dropdown is empty. The 'PARTITION definition' field is empty. At the bottom, there is a 'Save' button, followed by 'Or Add', a text input field containing '1', and 'column(s)', and a 'Go' button.

Gambar 2.8
Tampilan Menyimpan Tabel

7. Tampilan struktur tabel yang telah dibuat.



The screenshot shows the phpMyAdmin interface with the 'kegiatan' table selected. The table structure is as follows:

#	Column	Type	Collation	Attributes	Null	Default	Extra	Action
1	id	int(11)			No		AUTO_INCREMENT	Change Drop More
2	id_sdpd	int(11)			No			Change Drop More
3	nama_jaksa1	varchar(50)	latin1_swedish_ci		No			Change Drop More
4	nip1	varchar(15)	latin1_swedish_ci		No			Change Drop More
5	pangkat1	varchar(30)	latin1_swedish_ci		No			Change Drop More
6	jabatan1	varchar(50)	latin1_swedish_ci		No			Change Drop More
7	nama_jaksa2	varchar(50)	latin1_swedish_ci		No			Change Drop More
8	nip2	varchar(15)	latin1_swedish_ci		No			Change Drop More
9	pangkat2	varchar(30)	latin1_swedish_ci		No			Change Drop More
10	jabatan2	varchar(50)	latin1_swedish_ci		No			Change Drop More
11	nama_jaksa3	varchar(50)	latin1_swedish_ci		No			Change Drop More
12	nip3	varchar(15)	latin1_swedish_ci		No			Change Drop More
13	pangkat3	varchar(30)	latin1_swedish_ci		No			Change Drop More
14	jabatan3	varchar(50)	latin1_swedish_ci		No			Change Drop More

Gambar 2.9
Tampilan Struktur Tabel